

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

**Informační systém snímkového
materiálu**
Information System of Snapshot Material

2011

Radek Ryba

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

Dne 6. 5. 2011

.....
Radek Ryba

Poděkování

Děkuji vedoucímu bakalářské práce doc. Ing. Lačezaru Ličevu, CSc. za účinnou odbornou pomoc a cenné rady při zpracování. Dále bych chtěl poděkovat své rodině za podporu při studiu.

Abstrakt

Bakalářská práce je zaměřena na návrh a realizaci informačního systému snímkového materiálu. V první části práce se zaměřuji na požadavky informačního systému. Druhá část se již věnuje analýze informačního systému, obsahuje ER diagram, datový slovník, seznam požadovaných funkcí. Třetí část se zaměřuje na implementaci informačního systému, pojednává o výběru použité technologie, popisuje vývojová prostředí a praktickou realizaci. Poslední kapitola obsahuje závěrečné zhodnocení dosažených výsledků a shrnuje učiněné poznatky. Cílem této bakalářské práce je ověřit si vědomosti získané během studia a vytvořit funkční informační systém.

Klíčová slova: Informační systém, snímkový materiál, ASP.NET, C#, databáze, SQL, SQL Server, Visual Studio

Abstract (English)

This bachelor thesis is focused on the design and realization of information system of snapshot material. In the first part of thesis I'm focused on the system requirements. The second part is already about analysis of the system, contains ER diagram, the data dictionary and a list of required functions. The third part is focused on the implementation of information system, is about choice of technology, describes development environments and practical realization. The last part contains the final evaluation of the results and recapitulate downright findings. The aim of this work is to verify the knowledge acquired by studies and create a functional information system.

Keywords: Information system, snapshot material, ASP.NET, C#, database, SQL, SQL Server, Visual Studio

Seznam použitých symbolů a zkratek

ADO.NET – ActiveX Data Objects

ASP – Active Server Pages

ASP.NET – nástupce technologie ASP (Active Server Pages)

CLR – Common Language Runtime

CLS – Content Management Systems

DFD – Data flow diagram

ERD – Entity Relationship Diagrams

IIS – Internet Information Services

LINQ – Language Integrated Query

SQL – Structured Query Language

HTML – Hyper Text Markup Language

HTTP – Hyper Text Transfer Protocol

Obsah

1 Úvod.....	1
2 Požadavky informačního systému	2
2.1 Specifikace funkčních požadavků	2
2.2 Specifikace nefunkčních požadavků	6
3 Analýza informačního systému.....	7
3.1 Datová analýza	7
3.1.1 Lineární zápis typů entit	8
3.1.2 E-R diagram	8
3.1.3 Datový slovník	9
3.2 Funkční analýza	10
4 Implementace	20
4.1 Výběr technologie	20
4.2 Instalace	21
4.3 Výsledný systém	23
5 Závěr	25
Literatura.....	26
Přílohy	27

1 Úvod

V dnešní době se s informačními systémy setkáváme snad na každém kroku, ať už se jedná o obří firmy, internetové obchody, či stránky nadšenců. Téměř na vše již existuje alespoň jeden informační systém, avšak informační systém snímkového materiálu najdete jen stěží. Proto se v této bakalářské práci zabývám tvorbou tohoto systému.

Snímkový materiál se vyskytuje ve spoustě možných odvětví, setkáváme se s ním například v medicíně, důlním průmyslu, astronomii, atd.

Pro takovýto systém potřebujeme nějakým způsobem komunikovat s uživatelem a serverem. K tomuto se využívají webové aplikace, obsahující stránky s dynamicky generovaným obsahem. Komunikace s uživatelem zde probíhá prostřednictvím protokolu HTTP. Tento typ komunikace užívají softwary typu klient/server, kdy klient (zpravidla webový prohlížeč) zasílá serveru jednotlivé požadavky a obdržené webové stránky zobrazuje.

Prostředků pro vývoj webových aplikací, je mnoho, díky rozmanitosti různých technologií. Pro účely této práce jsem si vybral podle mě nejvhodnější technologii ASP.NET. ASP.NET je jedna z nejmladších objektových technologií, která využívá velmi obsáhlý .NET Framework, obsahující různé již odladěné nástroje a postupy vývoje webových aplikací. Je založen na CLR (Common Language Runtime), který je sdílen všemi aplikacemi postavenými na .NET Frameworku. Díky tomu můžeme realizovat své projekty v jakémkoliv jazyce podporujícím CLR, např. Visual Basic.NET, JScript.NET, C#, atd. Tyto jazyky však musí splňovat určité předpoklady dané CLS (Common Language Specification), obsahující soubory vlastností a požadavků kladených na tyto jazyky. Pro tuto práci jsem si zvolil jazyk C#, jelikož je mně nejbližší a nejčastěji jsem s ním pracoval.

Webové stránky jsou tvořeny objekty, ovládacími prvky (Controls), které jsou obdobné ovládacím prvkům ve Windows aplikacích. Umožňuje nám to tedy používat ovládací prvky jako tlačítko (Button), nápis (Label) a další. Samozřejmostí je přiřazování určitých vlastností těmto prvkům, zachytávání na nich vyvolaných událostí, atd.

Abychom však mohli začít vytvářet takovouto webovou aplikaci, musíme si nejdříve celý informační systém promyslet a navrhnout. Proto se v první části práce zaměřuji na požadavky informačního systému. V druhé části již analyzuji informační systém, vytvářím ER diagram, datový slovník, seznam požadovaných funkcí, atd. Třetí část se již věnuje samotné implementaci informačního systému. Obsahuje tvorbu tabulek, postup při vytváření systému, popisuje použitá vývojová prostředí, jak postupovat pro zprovoznění tohoto systému, atd. Nakonec zhodnotím dosažené výsledky a shrnu učiněné poznatky.

2 Požadavky informačního systému

Než se pustíme do samotného návrhu systému, musíme si určit co od tohoto systému očekáváme, co by měl umět a jaké na něj máme nároky.

To hlavní by se dalo sepsat takto:

- uživatelé – chceme, aby do systému nemohl přistupovat každý, ale jen registrovaní uživatelé
- projekty – registrovaní uživatelé by měli do systému přidávat své vlastní projekty obsahující snímky a další informace, tvar projektu by měl být jednotný např. jméno projektu, popis projektu, datum vytvoření, autor projektu, atd.
- snímky – možnost přidávat do vytvořeného projektu snímky registrovanými uživateli, sjednocený tvar snímku, jen určité druhy snímku
- kategorie – projekty by měly být rozděleny podle různých kategorií, možnost zobrazovat jen projekty ze zadané kategorie
- komentáře – umožnit uživatelům přidávat k daným snímkům komentáře, obsahující údaje jako uživatele, datum přidání, text komentáře
- administrace – možnost spravovat uživatele, jejich projekty, komentáře
- další funkce – rozšiřující funkce vznikající při návrhu a řešení projektu
- design systému – měli bychom si dopředu rozmyslet grafickou stránku projektu

2.1 Specifikace funkčních požadavků

Proč?

Potřebujeme vést evidenci snímkového materiálu, kde by se mohly evidovat projekty se snímky různých uživatelů, uživatelé by je mohli zobrazovat, komentovat. Zatím žádnou takovou evidenci nemáme.

Kdo?

Se systémem budou pracovat uživatelé a admin.

Uživatelé: vytváření nových projektů, komentování existujících

Admin: uživatel s většími právy, administrace celého systému

Vstupy?

U uživatelů budeme evidovat jednoznačný login uživatele a heslo, pod kterými se bude přihlašovat, jeho jméno, příjmení, email, informaci jestli se jedná o admina a datum posledního přihlášení do systému.

U projektů budeme evidovat jednoznačné číslo projektu, název projektu, kategorii, datum vytvoření projektu, popis projektu a autora projektu.

U snímků budeme evidovat jednoznačné číslo snímku, název snímku, popis snímku a projekt, ke kterému snímek patří.

U komentářů budeme evidovat jednoznačné číslo komentáře, autora komentáře, snímek, kterému byl komentář přidán, datum vzniku komentáře a text komentáře.

U každého snímku je uveden projekt, do kterého patří. Každý snímek patří do jednoho projektu, ale v jednom projektu může být více snímků.

U každého projektu je uveden uživatel, který projekt vytvořil. Každý projekt patří jednomu uživateli, ale jeden uživatel může mít více projektů.

U každého komentáře je uveden uživatel, který komentář vytvořil. Každý komentář patří jednomu uživateli, ale jeden uživatel může mít více komentářů.

U každého komentáře je uveden snímek, ke kterému patří. Každý komentář patří jednomu snímku, ale jeden snímek může mít více komentářů.

Výstupy?

Různé seznamy projektů.

Proj_id	Nazev	Kategorie	Datum	Uziv_id	Popis

Zobrazení či stažení různých snímku.

Proj_id

Snim_id
Nazev
Popis

Snim_id
Nazev
Popis

Snim_id
Nazev
Popis

Nazev bude obsahovat cestu ke snímku a jeho zobrazení

Zobrazení komentářů.

Kom_id	Snim_id	
Vložil:	Uziv_id	Datum
Text		

Seznam uživatelů, kteří se zajímají o snímkový materiál.

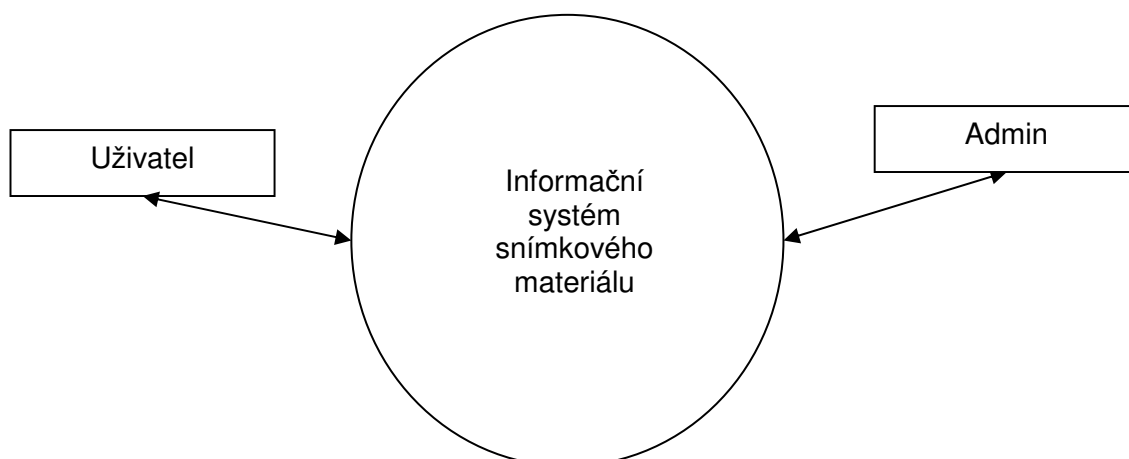
Uziv_id	Jmeno	Prijmeni	Email	Admin	Poslední_Prihlaseni

Funkce: seznam událostí a reakcí

událost	reakce	aktér
nový uživatel	zapiš do seznamu uživatelů	uživatel, admin
nový admin	uprav uživatele na admina	admin
nový projekt	zapiš do seznamu projektů	uživatel, admin
nový snímek projektu	zapiš do seznamu snímků	uživatel, admin
nový komentář snímku	zapiš do seznamu komentářů	uživatel, admin
výběr projektu	zobraz vybraný projekt	uživatel, admin
výběr uživatele	zobraz vybraného uživatele	admin
nežádoucí uživatel	smaž ze seznamu uživatelů	admin
nežádoucí můj projekt	smaž ze seznamu projektů	uživatel, admin
nežádoucí cizí projekt	smaž ze seznamu projektů	admin
nežádoucí můj komentář	smaž ze seznamu komentářů	uživatel, admin
nežádoucí cizí komentář	smaž ze seznamu komentářů	admin
nechtěné heslo	uprav v seznamu uživatelů	uživatel, admin

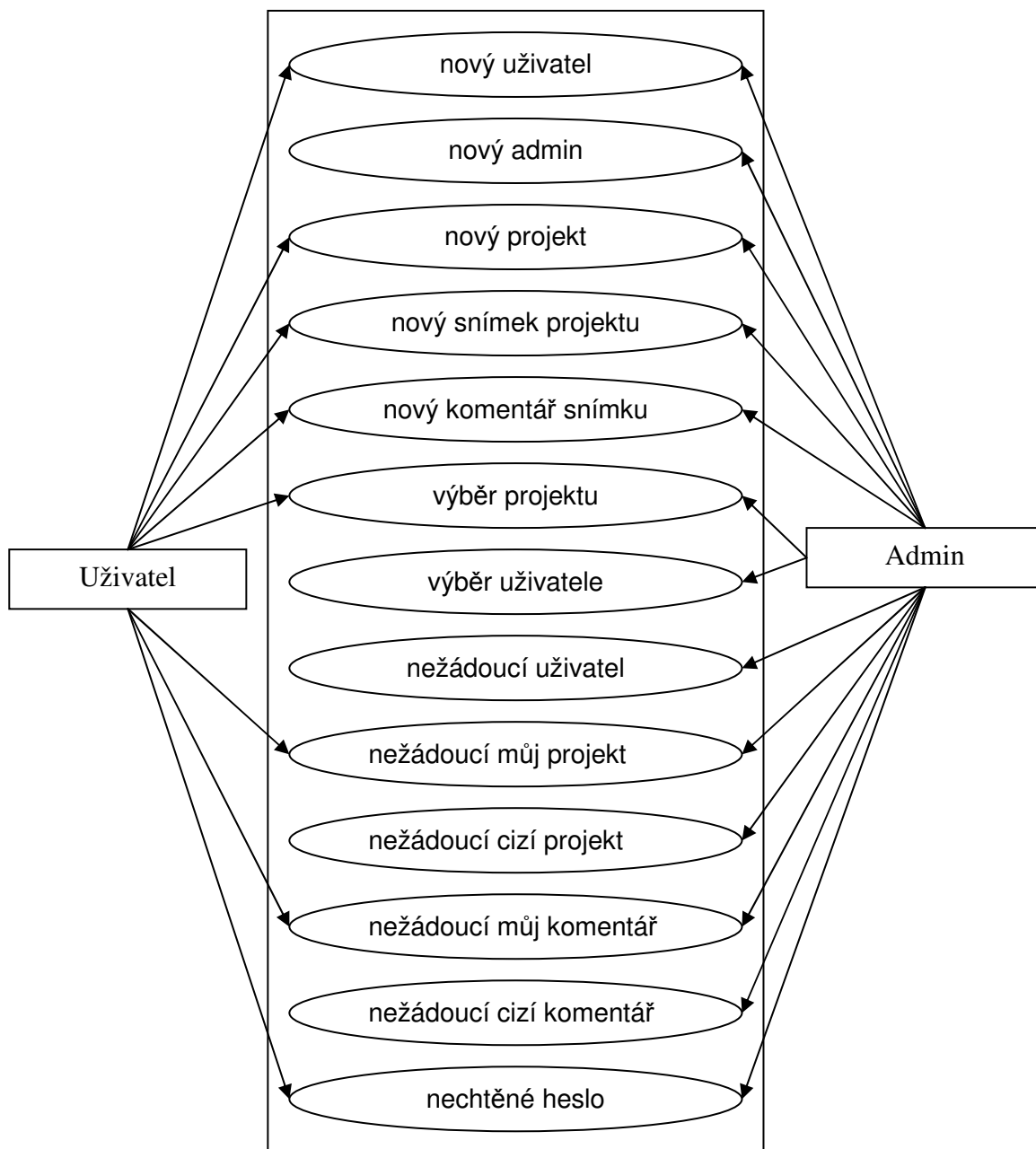
Okolí: kontextový diagram

Nejvyšší DFD, strukturuje okolí systému.



Okolí a funkce: model jednání

Slouží k propojení informace kontextového diagramu se seznamem událostí a reakcí, přiděluje události jednotlivým aktérům.



2.2 Specifikace nefunkčních požadavků

Požadavky na výsledný program:
efektivita, spolehlivost, přenositelnost a použitelnost.

3 Analýza informačního systému

Analýza je dalším krokem k vytvoření funkčního budoucího informačního systému. Jejím cílem je poznání problémů, jejich popis a vytváření modelů dříve, než začneme se samotným řešením. Modely se používají pro správné pochopení struktury a funkcí systému. Jsou hlavním podkladem pro další řešení.

Pro informační systém máme 3 typy analýzy, které se vytváří v daném pořadí:

- 1) datová analýza
- 2) funkční analýza
- 3) dynamická analýza

3.1 Datová analýza

Datová analýza slouží ke zpracování datového modelu. Datový model se používá pro popis schémat databáze na různých úrovních.

Datový model je přehled prostředků, které slouží pro

- charakteristiku datových struktur prostřednictvím typů entit
- přiřazení popisných atributů jednotlivým typům entit
- charakteristiku vazeb mezi daty pomocí typů vztahů
- charakteristiku integritních omezení k znázornění souladu s realitou.

Výsledkem datové analýzy je konceptuální model. Jedná se o schematický model části reality, o níž se povede evidence v budovaném IS. Konceptuální model musí používat jazyk, který bude dostatečně přesný pro modelování reality, jejích základních objektů a jejich atributů, vazeb mezi objekty a jejich vlastností.

Pro záznam struktury databáze na konceptuální úrovni se nejčastěji používá Chenův E-R model, který kombinuje textové formální zápisy, grafické zobrazení typů entit, atributů a vztahů mezi entitami i doplňující textové informace. Je určen k návrhu struktury databáze a k popisu hlavních vlastností dat v databázi uložených [1].

Výsledný konceptuální model tedy obsahuje:

- lineární zápis seznamu typů entit a jejich atributů
- úplný grafický tvar ERD
- datový slovník
- seznam dalších IO týkajících se entit a vztahů

3.1.1 Lineární zápis typů entit

Primární klíč, *cizí klíč*

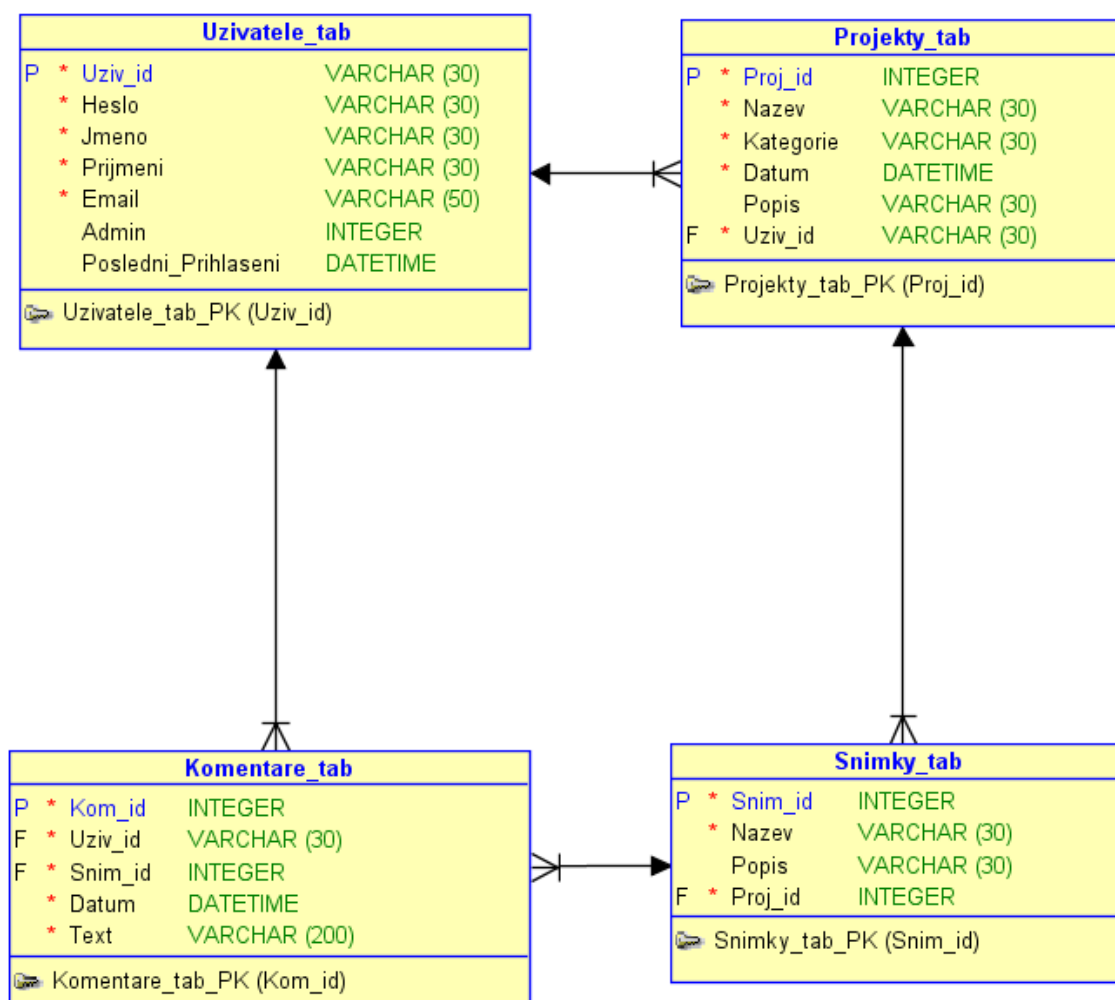
Uzivatele_tab (**Uziv_id**, Heslo, Jmeno, Prijmeni, Email, Admin, Posledni_Prihlaseni)

Projekty_tab (**Proj_id**, Nazev, Kategorie, Datum, Popis, *Uziv_id*)

Snimky_tab (**Snim_id**, Nazev, Popis, *Proj_id*)

Komentare_tab (**Kom_id**, *Uziv_id*, *Snim_id*, Datum, Text)

3.1.2 E-R diagram



3.1.3 Datový slovník

Uzivatele_tab

název	typ	velikost	klíč	null	popis
Uziv_id	VARCHAR	30	PK	ne	jednoznačná identifikace uživatele, slouží jako login
Heslo	VARCHAR	30	ne	ne	heslo pro přihlášení do systému
Jmeno	VARCHAR	30	ne	ne	jméno uživatele
Prijmeni	VARCHAR	30	ne	ne	příjmení uživatele
Email	VARCHAR	50	ne	ne	email uživatele
Admin	INTEGER		ne	ano	informace zda je uživatel admin
Posledni_Prihlaseni	DATETIME		ne	ano	datum posledního přihlášení do systému

Projekty_tab

název	typ	velikost	klíč	null	popis
Proj_id	INTEGER		PK	ne	jednoznačná identifikace projektu
Nazev	VARCHAR	30	ne	ne	název projektu
Kategorie	VARCHAR	30	ne	ne	kategorie, do které projekt patří
Datum	DATETIME		ne	ne	datum vytvoření projektu
Popis	VARCHAR	30	ne	ano	popis projektu
Uziv_id	VARCHAR	30	FK	ne	uživatel, který projekt vytvořil

Snimky_tab

název	typ	velikost	klíč	null	popis
Snim_id	INTEGER		PK	ne	jednoznačná identifikace snímku
Nazev	VARCHAR	30	ne	ne	název snímku
Popis	VARCHAR	30	ne	ano	popis snímku
Proj_id	INTEGER		FK	ne	projekt, do kterého snímek patří

Komentare_tab

název	typ	velikost	klíč	null	popis
Kom_id	INTEGER		PK	ne	jednoznačná identifikace komentáře
Uziv_id	VARCHAR	30	FK	ne	uživatel, který komentář vytvořil
Snim_id	INTEGER		FK	ne	snímek, kterému byl přidán komentář
Datum	DATETIME		ne	ne	datum vytvoření komentáře
Text	VARCHAR	200	ne	ne	text komentáře

3.2 Funkční analýza

Poté co jsme hotovi s návrhem struktury databáze, je třeba zaměřit se na funkce nad ní. Funkční analýza vychází ze seznamu funkčních požadavků, ze seznamu událostí a reakcí a z požadovaných vstupu a výstupů. Všechny tyto parametry máme již specifikovány v kapitole 2 Požadavky informačního systému, takže z nich můžeme vytvořit funkční model.

Funkční model by měl obsahovat tyto 2 úrovně:

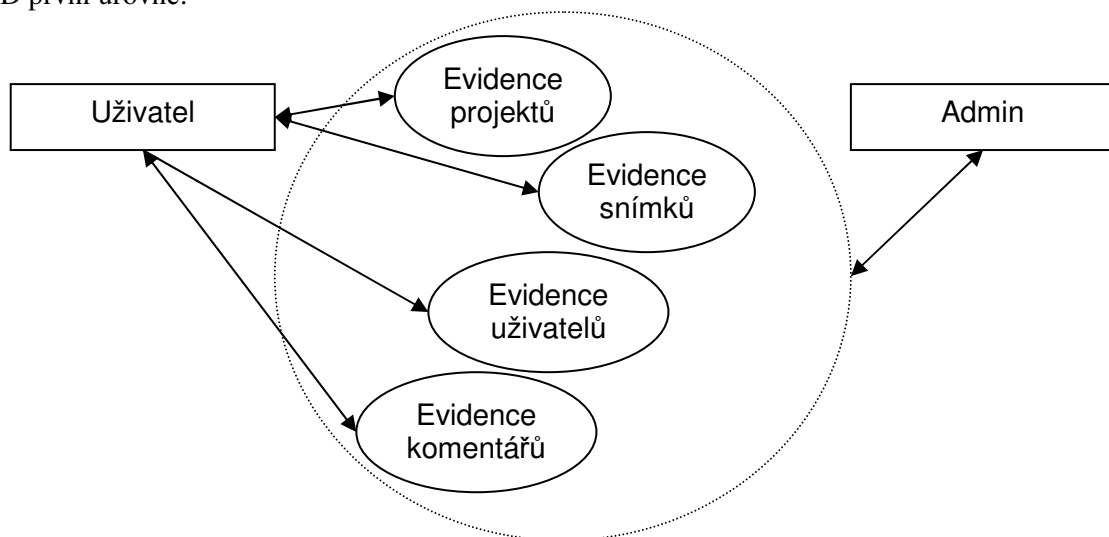
- vnější pohled je hrubý **grafický náhled na strukturu a hierarchii funkcí** systému
- vnitřní pohledy jsou podrobně rozpracované **algoritmy** (minispecifikace) pro jednotlivé akce.

Pomocí DFD vytvoříme první vnější pohled na funkce. DFD slouží pro návrh a zobrazení funkčního modelu systému. Měl by být dostatečně jednoduchý a názorný, někdy může sloužit i k upřesňování zadání [2].

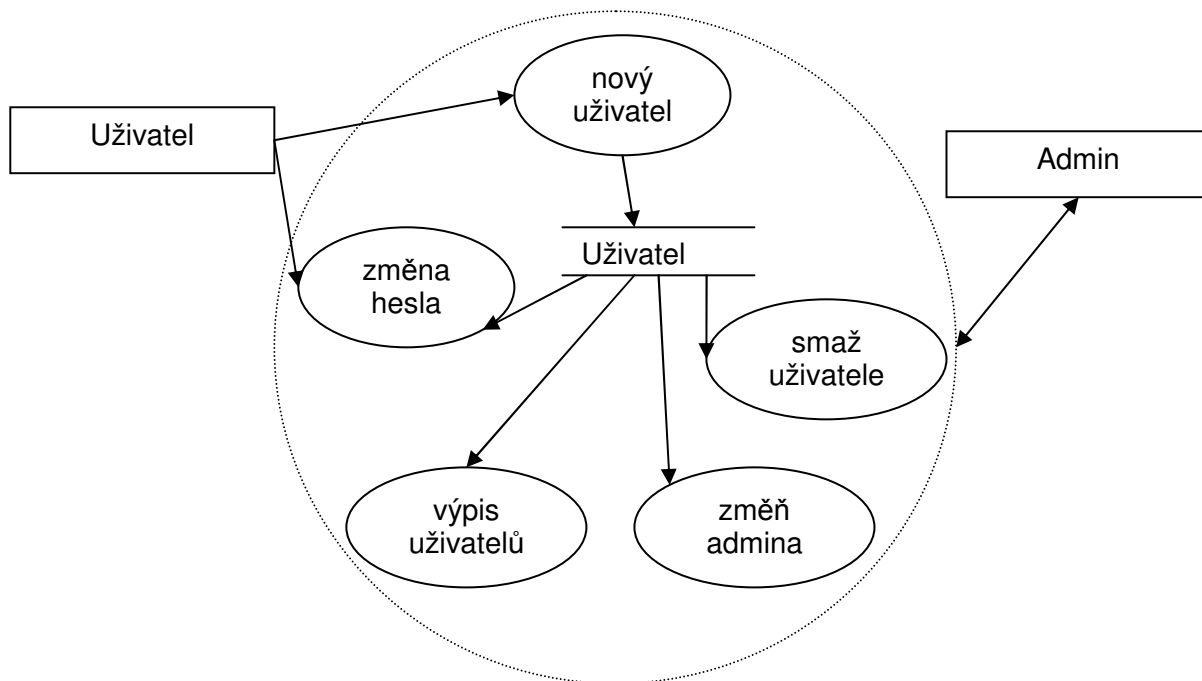
DFD nulté úrovně:



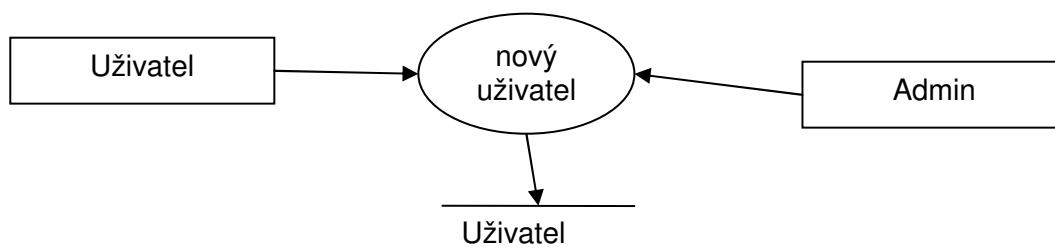
DFD první úrovně:



DFD – Uzivatele_tab – Evidence uživatelů

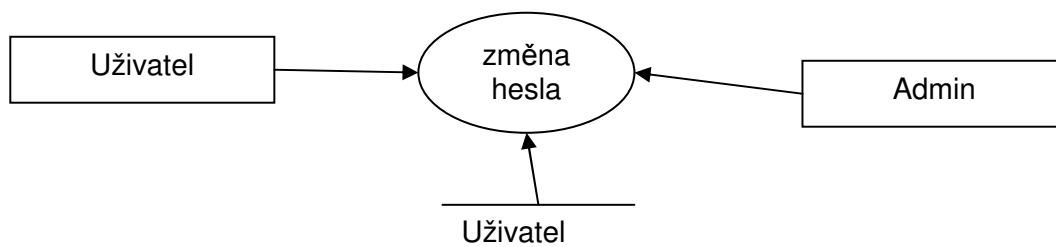


Minispecifikace – nový uživatel



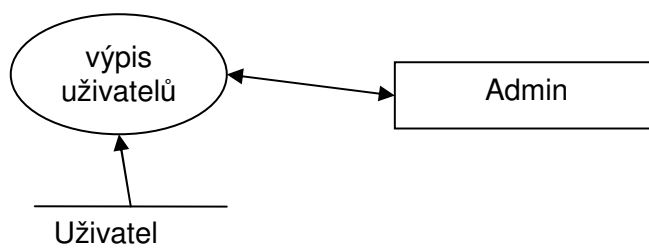
- 1) Zobraz formulář pro registraci
- 2) Vlož údaje do formuláře
- 3) Zkontroluj:
 - a) zda jsou vyplněny všechny povinné atributy
Pokud nejsou vypiš chybovou hlášku „Doplň chybějící atribut“ a vrať se na bod 1
 - b) zda zadané Uziv_id již není v systému:
Pokud Uziv_id existuje v systému vypiš chybovou hlášku „Uživatel již existuje“ a vrať se na bod 1
- 4) Ulož do tabulky Uzivatele_tab nový údaj z formuláře
- 5) Zobraz hlášku „Registrace byla úspěšná“

Minispecifikace – změna hesla



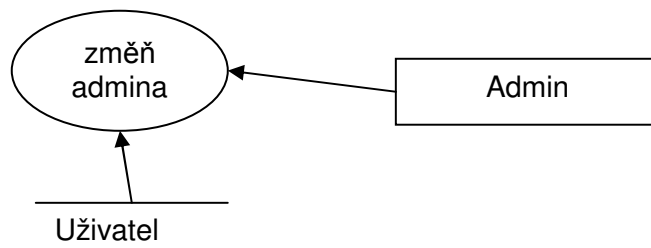
- 1) Zobraz formulář pro změnu hesla
- 2) Zkontroluj:
 - a) zda jsou vyplněny všechny povinné atributy
Pokud nejsou vypiš chybovou hlášku „Doplň chybějící atribut“ a vrať se na bod 1
 - b) zda staré heslo souhlasí
Pokud ne vypiš chybovou hlášku „Neplatné heslo“ a vrať se na bod 1
 - c) zda kontrola hesla a nové heslo souhlasí
Pokud ne vypiš chybovou hlášku „Kontrola hesla nesouhlasí“ a vrať se na bod 1
- 3) Uprav záznam v tabulce Uzivatele_tab
- 4) Zobraz hlášku „Změna proběhla úspěšně“

Minispecifikace – výpis uživatelů



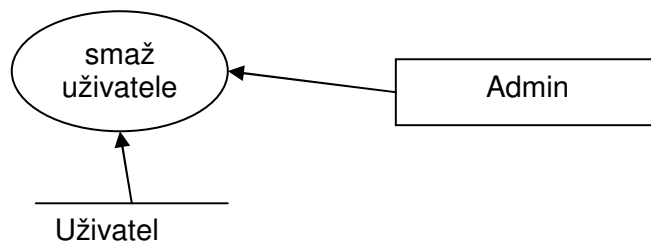
- 1) Zkontroluj zda přihlášený uživatel je admin
Pokud není přesměruj na stránku pro přihlášení
- 2) Zobraz stránku s uživateli
- 3) Nahraj všechny uživatele z tabulky Uzivatele_tab
- 4) Zobraz výstup pro seznam uživatelů

Minispecifikace – změň admina



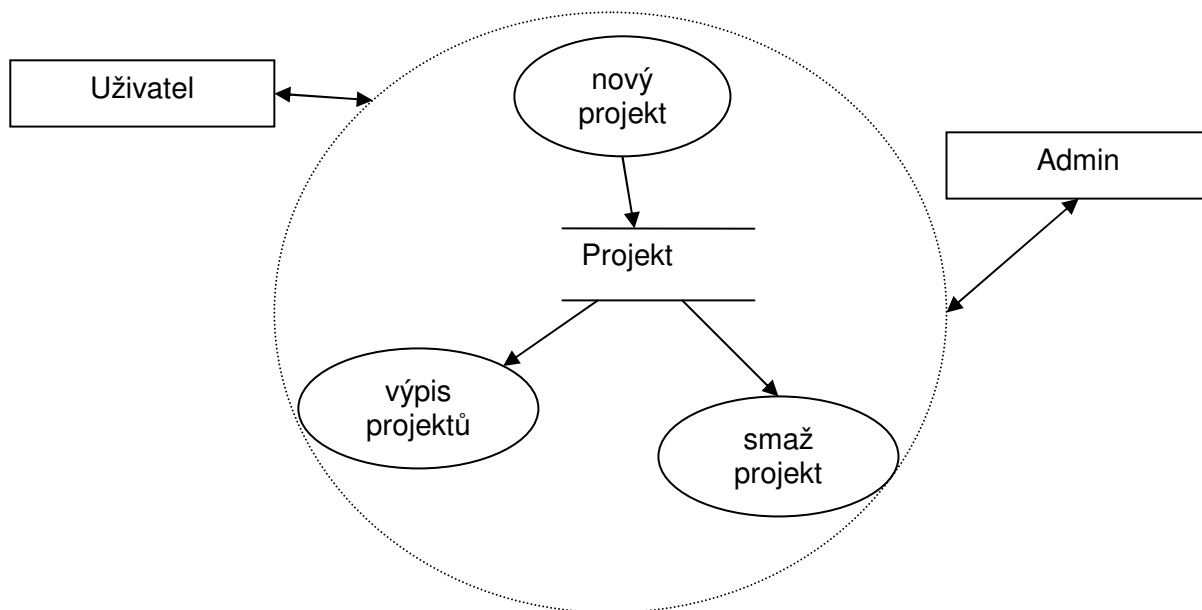
- 1) Proved' funkci výpis uživatelů
- 2) Vyber uživatele, u kterého chceme provést změnu
- 3) Zkontroluj zda vybraný uživatel je admin
 Pokud není nabídni možnost „Přidat admina“
 Pokud je nabídni možnost „Odebrat admina“
- 3) Uprav záznam v tabulce Uzivatele_tab
- 4) Zobraz hlášku „Změna provedena“

Minispecifikace – smaž uživatele

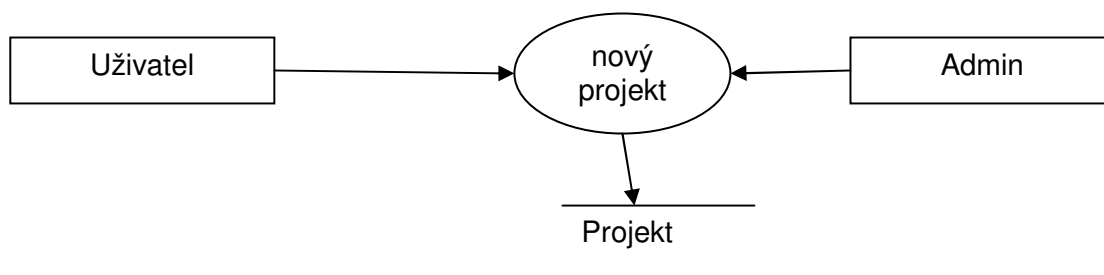


- 1) Proved' funkci výpis uživatelů
- 2) Vyber uživatele, kterého chceme smazat
- 3) Zobraz hlášku „Opravdu chcete smazat uživatele? ano/ne“
- 3) Je-li odpověď ano smaž záznam z tabulky Uzivatele_tab
- 4) Zobraz hlášku „Změna provedena“

DFD – Projekty_tab – Evidence projektů

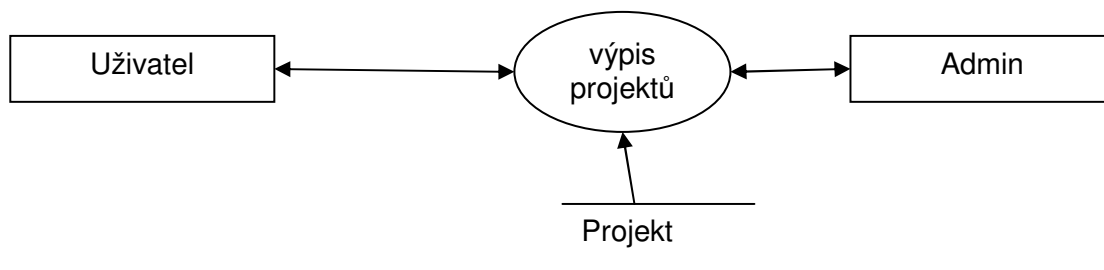


Minispecifikace – nový projekt



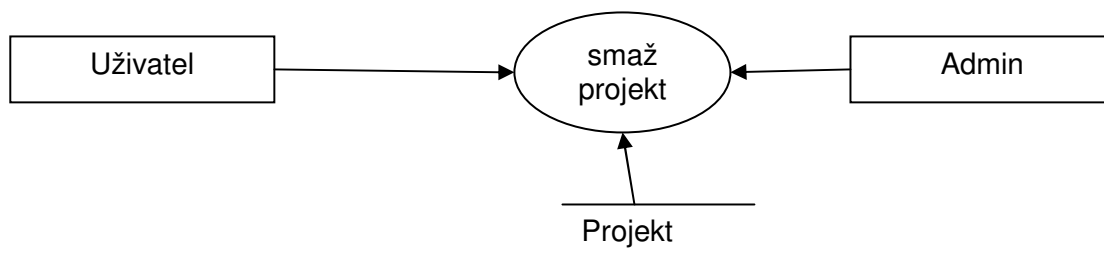
- 1) Zobraz formulář pro nový projekt
- 2) Vlož údaje do formuláře
- 3) Zkontroluj zda jsou vyplněny všechny povinné atributy
Pokud nejsou vypiš chybovou hlášku „Doplň chybějící atribut“ a vrať se na bod 2
- 4) Ulož do tabulky Projekty_tab nový údaj z formuláře
- 5) Zobraz hlášku „Projekt byl vytvořen, pokračujte vkládáním snímků“
- 6) Zobraz formulář pro vkládání snímků do projektu

Minispecifikace – výpis projektů



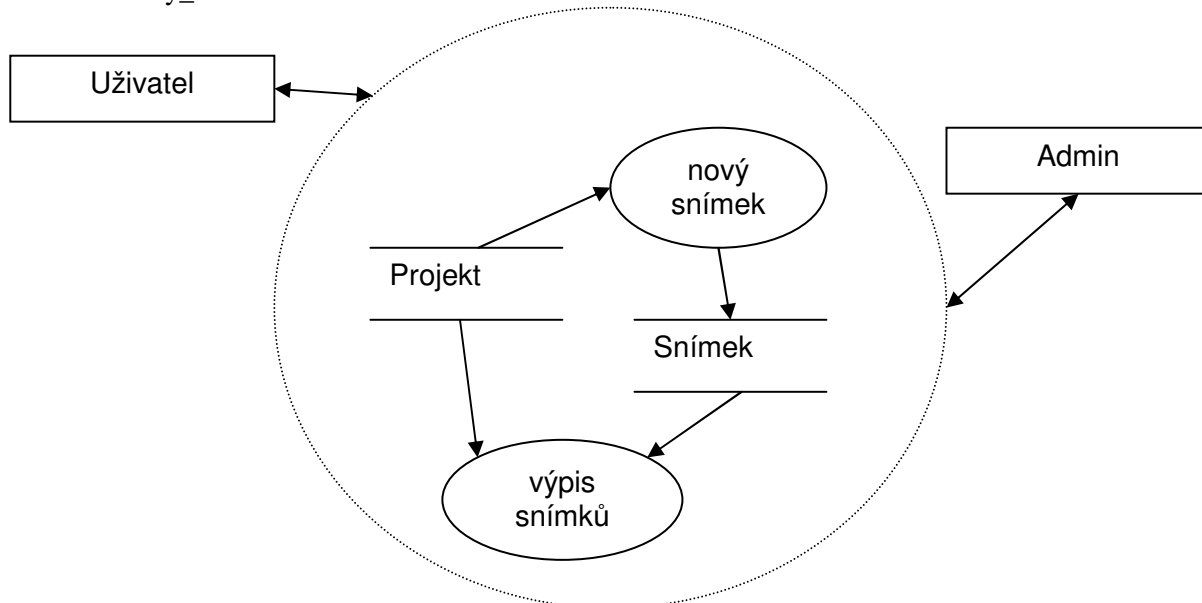
- 1) Zobraz stránku s projekty
- 3) Nahraj projekty podle výběru kategorie z tabulky Projekty_tab
- 4) Zobraz výstup pro seznam projektů

Minispecifikace – smaž projekt

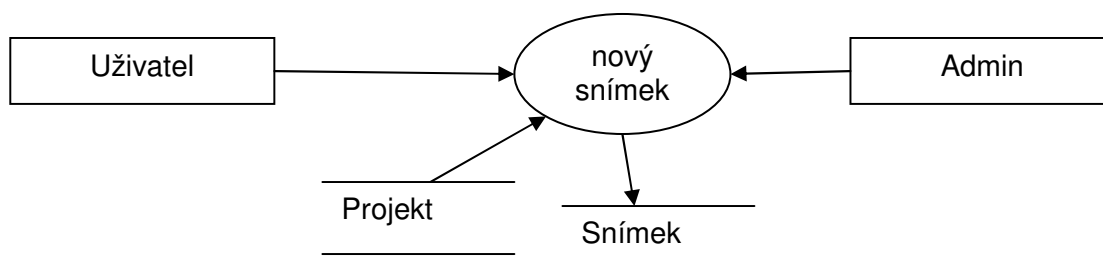


- 1) Proved' funkci výpis projektů
- 2) Vyber projekt, který chceme smazat
- 3) Zkontroluj vlastníka projektu s přihlášeným uživatelem
 - a) Pokud nejsou Uziv_id stejné nebo pokud není přihlášený uživatel admin, nenabídne možnost smazat projekt
 - b) Pokud jsou Uziv_id stejné nebo je přihlášený uživatel admin povolí možnost smazat projekt
- 3) Zobraz hlášku „Opravdu chcete smazat projekt? ano/ne“
- 4) Je-li odpověď ano smaž záznam z tabulky Projekty_tab
- 4) Zobraz hlášku „Projekt byl smazán“

DFD – Snímky_tab – Evidence snímků

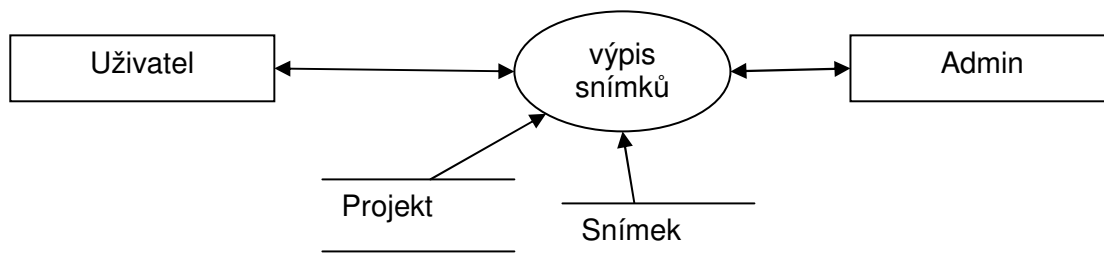


Minispecifikace – nový snímek



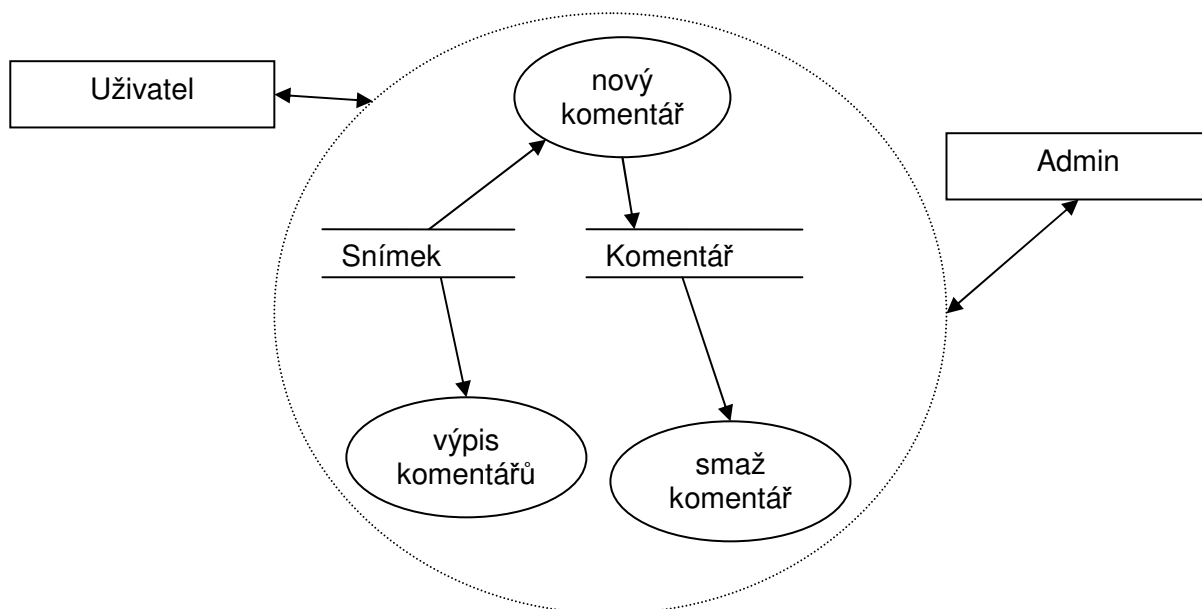
- 1) Zobraz formulář pro nový snímek po funkci nový projekt
- 2) Vlož údaje do formuláře
- 3) Zkontroluj zda jsou vyplněny všechny povinné atributy
Pokud nejsou vypiš chybovou hlášku „Doplň chybějící atribut“ a vrať se na bod 2
- 4) Ulož do tabulky Snímky_tab nový údaj z formuláře
- 5) Zobraz hlášku „Snímek byl nahrán,pokračujte dalším snímkem“
- 6) Zobraz formulář pro vkládání snímků do projektu
- 7) Pro nahrání dalšího snímku do projektu jdi na bod 2
- 8) Ukonči nahrávání snímku

Minispecifikace – výpis snímků

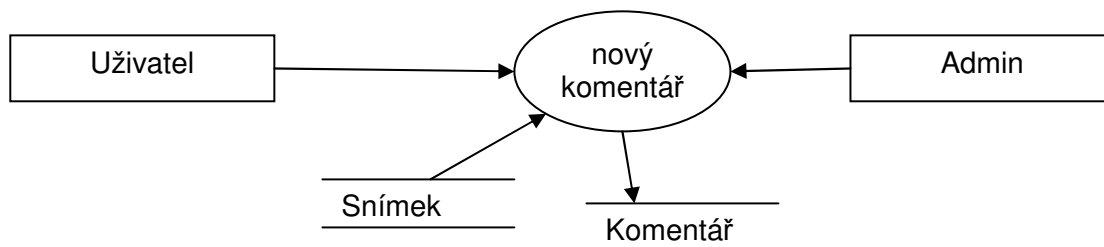


- 1) Proved' funkci výpis projektů
- 2) Vyber projekt jehož snímky chceš zobrazit
- 3) Ulož Projekty_tab.Proj_ID
- 4) Nahraj snímky, u kterých platí Projekty_tab.Proj_ID=Snimky_tab.Proj_ID z tabulky Snimky_tab
- 5) Zobraz výstup pro zobrazení snímků

DFD – Komentare_tab – Evidence komentářů

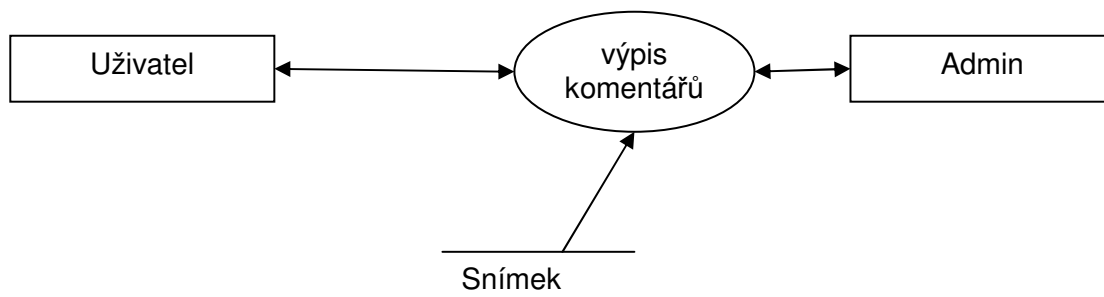


Minispecifikace – nový komentář



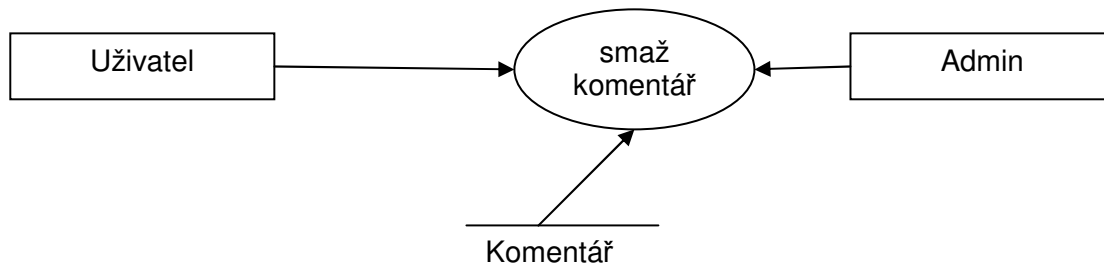
- 1) Proved' funkci výpis snímku
- 2) Vyber snímek, který chceš komentovat
- 3) Vlož údaje do formuláře pro nový komentář
- 3) Zkontroluj zda jsou vyplněny všechny povinné atributy
Pokud nejsou vypiš chybovou hlášku „Doplň chybějící atribut“ a vrať se na bod 3
- 4) Ulož do tabulky Komentare_tab nový údaj z formuláře

Minispecifikace – výpis komentářů



- 1) Proved' funkci výpis snímku
- 2) Vyber snímek, jehož komentáře chceš zobrazit
- 3) Ulož Snimky_tab.Snim_id
- 4) Nahraj komentáře, u kterých platí Snimky_tab.Snim_id=Komentare_tab.Snim_id z tabulky Komentare_tab
- 5) Zobraz výstup pro zobrazení komentaru

Minispecifikace – smaž komentář



- 1) Proved' funkci výpis komentářů
- 2) Vyber komentář, který chceš smazat
- 3) Zkontroluj vlastníka komentáře s přihlášeným uživatelem
 - a) Pokud nejsou Uziv_id stejné nebo pokud není přihlášený uživatel admin, nenabídne možnost smazat komentář
 - b) Pokud jsou Uziv_id stejné nebo je přihlášený uživatel admin povolí možnost smazat projekt
- 4) Zobraz hlášku „Opravdu chcete smazat komentář? ano/ne“
- 5) Je-li odpověď ano smaž záznam z tabulky Komentare_tab
- 6) Zobraz hlášku „Komentář byl smazán“

4 Implementace

Implementace je proces uskutečňování teoreticky stanovené myšlenky nebo projektu za účelem jejího dalšího použití. Implementaci předchází analýza zadání, plánování postupu a očekávaných výsledků. Nesoulad mezi předpokladem a skutečností může být způsoben chybou implementace nebo chybou samotné metody. Správná metoda je funkční nezávisle na způsobu implementace [4].

4.1 Výběr technologie

Pro implementaci tohoto informačního systému jsem si zvolil technologii ASP.NET. Tuto technologii řadíme do tzv. skriptovacích. Skriptovací technologie využívají vsuvek v HTML kódu, kdy je mezi těmito vsuvkami kód webové aplikace, který server zpracuje a vygeneruje dynamický výstup již v čistém HTML kódu prohlížeči.

ASP.NET je celkem nová technologie vytvořená firmou Microsoft, která slouží k tvorbě webových aplikací. Je považován za následovníka starší technologie ASP, která díky jednoduchému generování HTML kódu na straně serveru přinesla revoluci v tvorbě webových aplikací.

ASP.NET není nový programovací jazyk, jedná se o technologii, která tvoří součást systému .NET Framework. Podle Microsoftu se díky této technologii zásadně sníží doba návrhu i tvorby webových stránek. Při použití ASP.NET je vytvoření webové aplikace až 3x rychlejší, než kdybychom použili jakýkoliv jiný skriptovací jazyk. Díky tomu, že ASP.NET je založen na CLR, který je sdílen všemi aplikacemi postavenými na .NET Frameworku, můžeme své projekty realizovat v jakémkoliv jazyce, který CLR podporuje, např. Visual Basic.NET, JScript.NET, C#, Managed C++ a další.

Všechny tyto jazyky pracující na platformě .NET však musí splňovat určité předpoklady dané CLS, která obsahuje soubory vlastností a požadavků kladených na tyto jazyky. Díky předkompilování do několika DLL souborů jsou aplikace založené na ASP.NET rychlejší než ryze skriptovací jazyky, kde se stránky parsují při každém přístupu znovu.

.NET Framework stojí jako nadstavba nad operačním systémem. Na jeho CLR jsou postaveny všechny další knihovny a objekty, které slouží například pro přístup k datům (ADO.NET, XML, SQL), k vícevláknovému zpracování aplikací (Threading) k přístupu do sítě a internetu (NET) a k zpracovávání, auditování a monitorování bezpečnosti aplikací (Security).

Tato nová technologie nám tak ulehčuje přechod od programování klasických aplikací nejen pro Windows, ale i do prostředí webu. Webové stránky jsou tvořeny objekty, ovládacími prvky (Controls), které jsou obdobou ovládacích prvků ve Windows. Je tedy možné využívat ovládací prvky jako tlačítko, nápis a další i při tvorbě webových stránek. Samozřejmostí je přiřazování určitých vlastností těmto prvkům, zachytávání událostí, atd. Tyto webové ovládací prvky produkují HTML kód, který tvoří část výsledné stránky[4].

4.2 Instalace

.NET Framework

Pro činnost webových stránek v ASP.NET je zapotřebí komponenta zvaná Microsoft .NET Framework (nejnovější verze je .NET Framework 4).

Ten se stará o věci, které jsou již považovány jako samozřejmé a to :

- správa paměti, vytváření a rušení objektů
- spouštění a zastavování vláken kódu
- bezpečnost kódu a kontrola oprávnění k prováděným operacím
- natahování potřebných knihoven a komponent do paměti apod.

.NET Framework se dá stáhnout z oficiálních stránek firmy Microsoft [3].

SQL Server

V našem informačním systému potřebujeme zobrazit data z relační databáze. Webové stránky ASP.NET jsou schopny zobrazit data z jakékoli databáze, ke které je k dispozici ODBC nebo OLEDB ovladač [3].

Pro tuto bakalářskou práci jsem použil relační databázi Microsoft SQL Server 2008 R2, kterou lze stáhnout opět na stránkách Microsoft. Abych mohl databázi i spravovat je potřeba si stáhnout i Microsoft SQL Server Management Studio pro danou verzi SQL serveru. SQL Server je ke stažení zdarma, přesto nabízí většinu funkcí, kterých bylo zapotřebí při vývoji této práce.

SQL kód pro vytvoření tabulek a vložení admina přikládám jako přílohu pod název SQL.txt.

Webový server

Abychom mohli tento informační systém provozovat, je zapotřebí samozřejmě nějaký webový server. Hotový web musíme umístit na počítač s IIS (Internet Information Serverem).

IIS nám nabízí vše, co je k provozování webu potřeba, a to jak pro intranet tak i pro Internet. Samotné publikování webu na IIS server je velmi jednoduché [3]. Jelikož já samotný počítač s IIS, který by byl vhodný pro hostování našeho informačního systému nemám, musel jsem využít služby hosterů. Spousta z nich podporuje ASP.NET a SQL Server za měsíční poplatek, někteří dávají k dispozici omezené hostování aplikace zdarma. K otestování funkčnosti této bakalářské práce sem využil bezplatné hostování, které je pro vyzkoušení dostačující. Informační systém najdete na stránce <http://snimkovymaterial.asp2.cz/>.

Nasazení na vzdálený server

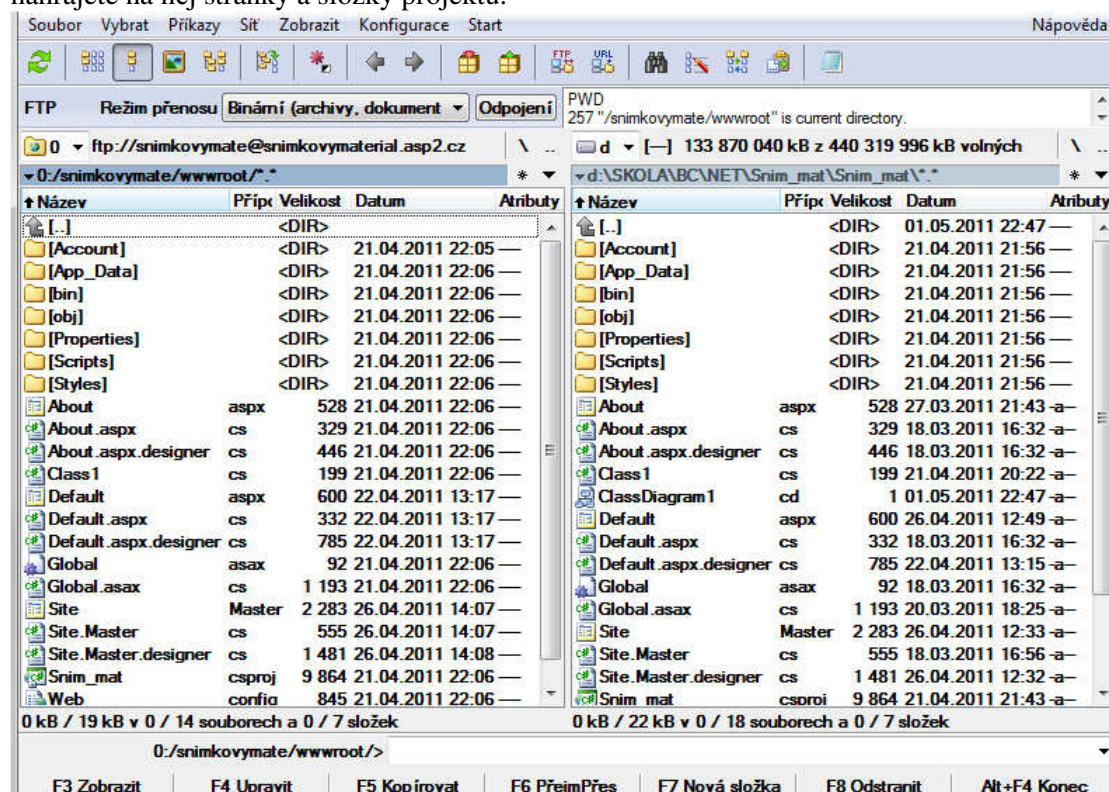
Po registraci dostanete u hostera typicky k dispozici adresu URL, jméno a heslo. Po přihlášení by jste si měli v administrační části zažádat o vytvoření MS SQL databáze. Po vytvoření obdržíte od hostera email s vypsanými parametry k jejímu připojení. Přihlaste se do databáze a použijte SQL kód v příloze SQL.txt k vytvoření tabulek a administrátora. Stačí zaměnit údaje v řádku za vaše :

```
insert into Uzivatele_tab(Uziv_id,Heslo,Jmeno,Prijmeni,Email,Admin)
values ('VasLogin','VaseHeslo','VaseJmeno','VasePrijmeni', 'VasEmail',1);
```

Poté stačí otevřít soubor Web.Config, který se nachází v adresáři projektu, například v poznámkovém bloku a upravit Data Source= podle vašeho SQL serveru:

```
<connectionStrings>
  <add name="Snim_matConnect"
connectionString="Data Source=VasServer.dbaserver.net;
Initial Catalog=VasServer;UID=VaseServerID;PWD=VaseServerHeslo;"
  providerName="System.Data.SqlClient" />
</connectionStrings>
```

Nakonec se pomocí libovolného FTP klientu připojíte na FTP server, který vám poskytl hoster a nahrajete na něj stránky a složky projektu.



4.3 Výsledný systém

Design

Pro design informačního systému sem využil Master Page technologie ASP.NET a jeho další komponenty. Pro ukázkou jak výsledný systém vypadá přikládám obrázek formu Projects.aspx, který se zpřístupní po přihlášení.

INFORMAČNÍ SYSTÉM SNÍMKOVÉHO MATERIÁLU

Log Out

Admin Fish

[Projekty v systému](#) [Nový projekt](#) [Změna hesla](#) [Uživatelé](#)

Kategorie:

[Důlní snímky](#)
[Filmy](#)
[Medicínské snímky](#)

	Proj. id	Nazev	Kategorie	Datum	Uživ. id	Popis
Select	1002	Důlní snímky1	Důlní snímky	21.4.2011 22:08:38	Fish	snímky z dolů
Select	1003	Medicínské snímky1	Medicínské snímky	21.4.2011 22:10:45	Fish	snímky z medicíny
Select	1004	filmy	Filmy	21.4.2011 22:17:12	user1	avi soubor

Nahoře vidíme Master Page, vlevo po něm login přihlášeného uživatele a informaci, jestli se jedná o admina či nikoliv. Dále zde máme linkbuttony pro práci v systému. Linkbutton Uživatelé je aktivní jen tehdy, jedná-li se o admina. Vlevo je datalist kategorie, který zobrazuje všechny kategorie v systému, po kliknutí zobrazí projekty jen s vybranou kategorií.

Zobrazení snímků v projektu Galerie.aspx:

INFORMAČNÍ SYSTÉM SNÍMKOVÉHO MATERIÁLU

Log Out

Fish

Projekt ID: 1003


snímek1


snímek2


snímek pdf

[Zpět](#)

Zobrazí snímky vybraného projektu, po kliknutí na snímek zobrazí zvětšeninu snímku, komentáře a možnost stažení snímku.

Programové řešení

ASP.NET nabízí spoustu možností, jak komunikovat s databází. Můžeme použít ovládací prvky zdroje dat, které umožňují zapouzdřit přístup k datům v ovládacím prvku, který můžete nakonfigurovat informacemi o připojení a dotazech, nebo také můžeme sami zapsat kód provádějící přístup k datům pomocí tříd technologie ADO.NET nebo LINQ dotazy.

Ovládací prvek `LinqDataSource` nám umožňuje použití syntaxe LINQ v kódu značek pro přístup k datům z různých zdrojů dat. Ovládací prvek `ObjectDataSource` pracuje s obchodním objektem ve střední vrstvě, s jehož pomocí načítá a upravuje data. Ovládací prvek `SqlDataSource` umožňuje poskytnout připojení ke zdroji dat a příkazy SQL pro načtení a úpravu dat. Já jsem se rozhodl použít ovládací prvek `SqlDataSource`. Zprostředkovatele a informaci o připojovacím řetězci definuji v souboru `Web.config`. To mně umožňuje opětovně používat informace s více instancemi ovládacích prvků dat [5].

Při loginu ověřuji, zda se z databáze vybere přesně jeden záznam se zadaným loginem a heslem. Pokud se počet vybraných záznamů nerovná 1 systém vyhodí chybovou hlášku. Když je počet záznamů roven 1, systém uloží do session `userID` a zkontroluje, zda se jedná o admina. Je-li uživatel admin nastaví se i session `admin`, jinak se nenastaví. Poté se nastaví `AuthCookie` a přejdu na form `Projects.aspx`. Zde se podle práv upraví rozhraní a `SqlDataSource` načte data podle zadaného `select` commandu. Jako admin můžete mazat všechny projekty a vidíte uživatele v systému, které můžete spravovat. Jako obyčejný uživatel můžete mazat jen své projekty a uživatele nevidíte. Můžete si vybrat zda chcete vidět všechny projekty, nebo jen projekty podle zadané kategorie. Při výběru projektu přejdete do jeho galerie na form `Galerie.aspx`. Další možnosti jsou nový projekt a změna hesla. Nový projekt přejde na nový form, kde vyplníte údaje a projekt se vloží do databáze. Objeví se komponenty pro přidávání snímků do právě vytvořeného projektu. Zde se zadává snímek, u kterého se kontroluje koncovka a generuje jeho název podle roku, měsíce, dne, hodiny a sekundy kvůli unikátnímu názvu. Poté je možné přidat popis, který se bude zobrazovat u snímku. Po nahrání všech snímků na server kliknete na tlačítko `dokončit`, které vás přesměruje opět na form pro projekty. Zde již vidíte váš nový projekt i novou kategorii. Po jeho vybrání se vám zobrazí možnost otevřít projekt, a pokud jste jeho vlastník tak i možnost smazat projekt. Při smazání se vás systém ještě jednou zeptá, zda jste si opravdu jistí a poté projekt smaže. Při otevření projektu se jeho ID uloží do session a na následující stránce se do `SqlDataSource` načtou snímky vybraného projektu. `Datalist` pak tyto snímky zobrazí pomocí itemu `imagebutton`, u něhož funkce `ImageUrl` určí zda se jedná o obrázek, video nebo pdf a načte požadovaný obrázek. Po kliknutí na `imagebutton` se uloží ID jeho snímku a pošle na další form `Snimek.aspx`. Zde se nahrají komentáře podle zadaného ID do připraveného `datalistu`, který je zobrazí. Opět se zjistí koncovka souboru funkcí `ImageUrl` a zobrazí se požadovaný snímek, jeho popis a tlačítko na stažení. U snímku můžeme mazat jen komentáře, které jsou naše, cizí komentáře může mazat jen admin. Když je snímek stažen a okomentován, pomocí tlačítek zpět se můžeme vrátit na požadovanou stránku. Pro ukončení práce se systémem slouží tlačítko `Log Out` v Master Page, které vymaže uložené sessiony a vrátí vás na stránku pro přihlášení. Více informací naleznete v programátorské dokumentaci.

5 Závěr

Ve zpracované bakalářské práci řeším Informační systém snímkového materiálu. Myslím, že hlavním přínosem této práce bude zefektivnění práce osob, které se pohybují v oblasti snímkového materiálu. Nejprve bylo nutné se detailněji seznámit se snímkovým materiálem a s požadavky, které by měl požadovaný systém řešit. Z těchto poznatků pak vychází samotný návrh systému.

Informační systém jsem naimplementoval v SQL Serveru 2008 R2 a v Microsoft Visual Studiu 2010 technologií ASP.NET. Ve výsledném informačním systému pracují 2 aktéři, a to přihlášení uživatelé a admini. Hlavní admin je vytvořen již při tvorbě tabulek, uživatelé jsou tvořeni pomocí registrace na webové aplikaci. Admin má na rozdíl od uživatelů práva na mazání všech projektů, komentářů i uživatelů, také může vytvářet další adminy. Další funkce má téměř totožné s uživateli. Uživatelé mohou nahrávat vlastní projekty, prohlížet cizí projekty, komentovat je, popřípadě stahovat jejich snímky. Mazat komentáře a projekty mohou jen tehdy, jsou-li jejími vlastníky.

Samotný návrh i implementace systému se podle mého zdařil, myslím, že všechny požadavky systému byly splněny. Hlavním přínosem pro mě bylo určitě bližší seznámení s tvorbou informačních systému, zefektivnění práce s technologií ASP.NET a přiblížení se k problematice umístění informačního systému na vzdálený server. Výsledný informační systém je nasazen na adrese <http://snimkovymaterial.asp2.cz/>

Bakalářskou práci by bylo možno rozšířit například o zasílání informačních emailů při komentování snímku, atd. Toto jsem ovšem neřešil, jelikož jsem neměl k dispozici smtp server. Další vhodné rozšíření by bylo určitě banovací systém, který by si uchovával IP adresy nepohodlných uživatelů. Případné další požadavky na přání uživatele by se také daly doplnit.

Literatura

[1] ŠARMANOVÁ, Jana. *Teorie zpracování dat* [online]. druhé, přepracované. Ostrava: VŠB – Technická univerzita Ostrava, 2007 [cit. 2010-11-09]. Dostupné z WWW: <<http://www.elearn.vsb.cz/archivcd/FEI/TZD/TZD.pdf>>. ISBN 978-80-248-1498-8

[2] ŠARMANOVÁ, Jana. *Databázové a informační systémy* [online]. první. Ostrava: VŠB – Technická univerzita Ostrava, 2007 [cit. 2010-11-09]. Dostupné z WWW: <<http://www.elearn.vsb.cz/archivcd/FEI/DAIS/DAIS.pdf>>. ISBN 978-80-248-1499-5

[3] MICROSOFT. *Webový vývoj v ASP.NET 2.0 pomocí bezplatných Express nástrojů* [online]. 2005 [cit. 2011-05-02]. Dostupné z WWW: < <http://msdn.microsoft.com/cs-cz/dd727769.aspx> >.

[4] *Wikipedie, otevřená encyklopedie* [online]. 2003 [cit. 2011-05-02]. Dostupné z WWW: <<http://cs.wikipedia.org>>.

[5] MSDN [online]. 2011 [cit. 2011-05-02]. Dostupné z WWW: < <http://msdn.microsoft.com> >.

[6] *DaniWeb IT Discussion Community* [online]. 2010 [cit. 2011-05-02]. Dostupné z WWW: < <http://www.daniweb.com/> >.

[7] *Databáze Education* [online]. 2010 [cit. 2011-05-02]. Dostupné z WWW: < <http://dbedu.cs.vsb.cz/> >.

Přílohy

Příloha č. 1: CD

Obsah CD

Na přiloženém CD naleznete:

- Bakalářskou práci
- Uživatelskou dokumentaci
- Programátorskou dokumentaci
- Soubor s SQL kódem SQL.txt
- Zdrojové kódy systému Snim_mat.rar